REMARKS/ARGUMENTS

Claims 1, 6-15, and 19 are pending. By this amendment, claims 1, 9, 11, and 15 are amended. Support for the amendment can be found at least at page 7, line 23, to page 8, line 18 and page 9, line 23 to page 10, line 11 of the specification. No new matter is introduced. Reconsideration and prompt allowance of the claims is respectfully requested.

35 U.S.C. § 112 Rejections

Claim 11 is rejected under 35 U.S.C. § 112 because of an informality. Claim 11 has been amended to remove the informality. Withdrawal of the claim rejection of claim 11 is respectfully requested.

35 U.S.C. § 103 Rejections

Claims 1, 6-16, and 19 are rejected under 35 U.S.C. § 103 (a) as being unpatentable over U.S. Patent 6,282,581 to Moore et al. (hereafter "Moore") in view of "Java Remote Method Invocation Specification (hereafter "Java RMI"). This rejection is respectfully traversed.

Moore is directed to a communications framework that is operable to support remote method invocation in a distributed object environment. As noted in the May 2, 2005 Response, Moore's method focuses on the binding and marshalling processes that enable a RMI process to function. However, Moore's method only reports a binding failure to the client requesting a process from the RMI server. In other words, Moore's method does not attempt to inform the process's interface of the binding failure and does not provide error recovery of such failure without restarting the process that is bound with an inactive RMI process.

Java RMI describes basic features of the RMI. Java RMI discusses, at sections 2.7, 4.3, 6.11, A.4, that a client process can recover from a failure by trapping an exception thrown by the RMI server to reconnect to the server. However, Java RMI does not teach or suggest how a process that is not bound with any active RMI processes can recover. Specifically, similar to Moore, Java RMI does not notify the process's interface that the interface is no longer bound with any active RMI processes, let alone recovering from such error without restarting the process that is bound with an inactive RMI process.

Contrary to Moore and Java RMI, amended claim 1 recites: "determining if the first RMI process is active and if the interface object is bound with any active RMI processes, wherein if the first RMI process is not active and if the interface object is not bound with any active RMI processes, an error occurs ... informing the interface object of the error when the monitoring agent determines that the interface object is not bound with any active RMI

processes; and rebinding the interface object with an active RMI process when the monitoring agent determines that the interface object is not bound with any active RMI processes, thereby recovering from the error without restarting the parent process that is bound with the first RMI process," (emphasis added). Claim 1 is amended to more precisely recite the novel features of the present application. When the parent process's interface object is not bound with any active RMI processes (e.g., the first RMI process that is bound with the interface object becomes inactive), the parent process's interface object is informed of such error and the parent process's interface object is rebound with an active RMI without restarting the parent process. As noted above, Moore and Java RMI, individually and in combination, do not disclose or suggest these features of amended claim 1. Therefore, amended claim 1 is allowable.

Claims 6-8 are allowable at least because they depend from allowable claim 1 and for the additional features they recite.

With respect to claim 9, for the same reason as discussed with respect to claim 1, Moore does not disclose or suggest "determining if the first RMI process is active and if the interface object is bound with any active RMI processes, wherein if the first RMI process is not active and if the interface object is not bound with any active RMI processes, an error occurs ... informing the interface object of the error when the monitoring agent determines that the interface object is not bound with any active RMI processes; and rebinding the interface object with an active RMI process when the monitoring agent determines that the interface object is not bound with any active RMI processes, thereby recovering from the error without restarting the parent process that is bound with the first RMI process," as recited in amended claim 9 (emphasis added). Therefore, claim 9 is allowable.

Claims 10-12 are allowable at least because they depend from allowable claim 9 and for the additional features they recite.

With respect to claim 15, for the same reason as discussed with respect to claim 1, Moore does not disclose or suggest "performing a list call to active RMI processes to determine whether the interface object is bound with any active RMI processes ... wherein if the interface object is not bound with any active RMI processes, an error occurs; 2) informing the interface object of the error when the monitoring agent determines that the interface object is not bound with any active RMI processes; 3) performing a rebind call to an active RMI process if the monitoring agent determines that the interface object is not bound with any active RMI processes, thereby recovering from the error without restarting the parent

process that is bound with the first RMI process," as recited in amended claim 15. Therefore, claim 15 is allowable.

Claims 16 and 19 are allowable at least because they depend from allowable claim 15 and for the additional features they recite.

Withdrawal of the rejection of claims 1, 6-16, and 19 under 35 U.S.C. §103 (a) is respectfully requested.

In view of the above remarks, Applicant respectfully submits that the application is in condition for allowance. Prompt examination and allowance are respectfully requested.

Should the Examiner believe that anything further is desired in order to place the application in even better condition for allowance, the Examiner is invited to contact Applicant's undersigned representative at the telephone number listed below.

Respectfully submitted,

Date: **October 7, 2005**

Kelly T. Lee
Registration No. 47,743
Andrews Kurth LLP
1701 Pennsylvania Ave, N.W.
Suite 300
Washington, DC 20006
Tel. (202) 662-2736
Fax (202) 662-2739